I CLAIM:

1. A data processing apparatus, comprising:

a processor operable to execute instructions;

a prefetch unit operable to prefetch instructions from a memory prior to sending those instructions to the processor for execution, the prefetch unit being operable to determine for a prefetched instruction whether that prefetched instruction is an instruction flow changing instruction, and based thereon to determine a fetch address for a next instruction to be prefetched by the prefetch unit;

a return stack accessible by the prefetch unit and operable to hold one or more addresses; and

prediction logic operable, if the prefetched instruction is a conditional instruction, to predict whether that prefetched instruction will be executed by the processor, the prefetch logic being operable to determine the fetch address dependent on the prediction from the prediction logic;

in the event that the prefetched instruction is a first type of instruction flow changing instruction and is conditional, and the prediction logic predicts that that prefetched instruction will be executed, the prefetch logic being operable to determine as the fetch address an address obtained from the return stack.

2. A data processing apparatus as claimed in Claim 1, wherein the first type of instruction flow changing instruction is a procedure return instruction operable when executed to cause the processor to return from a procedure being executed by the processor.

3. A data processing apparatus as claimed in Claim 1, wherein if the prefetch logic determines that the prefetched instruction is a second type of instruction flow changing instruction, the prefetch logic is further operable to determine a return address and to cause that return address to be placed on the return stack.

4. A data processing apparatus as claimed in Claim 3, wherein said second type of instruction flow changing instruction is a branch with link instruction, which is operable to identify a start address for a procedure to be executed by the processor, upon returning from the procedure the next instruction to be executed by the processor being specified by the return address.

5. A data processing apparatus as claimed in Claim 4, wherein the procedure is returned from by execution of one of said first type of instruction flow changing instructions.

6. A data processing apparatus as claimed in Claim 1, wherein said prediction logic is a dynamic prediction logic which is operable to provide a prediction as to whether the prefetched instruction will be executed by the processor dependent upon history information identifying an outcome of conditional instructions previously executed by the processor.

7. A data processing apparatus as claimed in Claim 1, wherein said prediction logic is provided within said prefetch unit.

8. A data processing apparatus as claimed in Claim 1, wherein said return stack is provided within said prefetch unit.

9. A data processing apparatus as claimed in Claim 1, wherein said prefetch unit comprises decode logic operable to determine for the prefetched instruction whether that prefetched instruction is an instruction flow changing instruction, and control logic operable in response to the decode logic to determine the fetch address for the next instruction to be prefetched by the prefetch unit.

10. A method of operating a data processing apparatus comprising a processor operable to execute instructions, a prefetch unit operable to prefetch instructions from a memory prior to sending those instructions to the processor for execution, and a return

stack accessible by the prefetch unit and operable to hold one or more addresses, the method comprising the steps of:

(a) determining for a prefetched instruction whether that prefetched instruction is an instruction flow changing instruction, and based thereon determining a fetch address for a next instruction to be prefetched by the prefetch unit;

(b) if the prefetched instruction is a conditional instruction, predicting whether that prefetched instruction will be executed by the processor, and at said step (a) determining the fetch address dependent on the prediction; and

(c) in the event that the prefetched instruction is a first type of instruction flow changing instruction and is conditional, and if said step (b) predicts that that prefetched instruction will be executed, determining as the fetch address an address obtained from the return stack.

11. A method as claimed in Claim 10, wherein the first type of instruction flow changing instruction is a procedure return instruction operable when executed to cause the processor to return from a procedure being executed by the processor.

12. A method as claimed in Claim 10, wherein if at said step (a) it is determined that the prefetched instruction is a second type of instruction flow changing instruction, the method further comprises the steps of:

determining a return address; and

placing that return address on the return stack.

13. A method as claimed in Claim 12, wherein said second type of instruction flow changing instruction is a branch with link instruction, which is operable to identify a start address for a procedure to be executed by the processor, upon returning from the procedure the next instruction to be executed by the processor being specified by the return address.

14. A method as claimed in Claim 13, further comprising the step of returning from the procedure by execution of one of said first type of instruction flow changing instructions.

15. A method as claimed in Claim 10, wherein said step (b) comprises the step of providing a prediction as to whether the prefetched instruction will be executed by the processor dependent upon history information identifying an outcome of conditional instructions previously executed by the processor.

16. A method as claimed in Claim 10, wherein said step (b) is performed within said prefetch unit.

17. A method as claimed in Claim 10, wherein said return stack is provided within said prefetch unit.

18. A method as claimed in Claim 10, wherein said prefetch unit comprises decode logic operable to determine for the prefetched instruction whether that prefetched instruction is an instruction flow changing instruction, and control logic operable in response to the decode logic to determine the fetch address for the next instruction to be prefetched by the prefetch unit.